Parallel and Distributed Machine Learning Algorithms for Scalable Big Data Analytics

Bal, Henri; Pal, Arindam

published in
Future Generation Computer Systems
2020

DOI (link to publisher)
10.1016/j.future.2019.07.009

document version
Publisher's PDF, also known as Version of record

Future Generation Computer Systems 108 (2020) 1159–1161

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Editorial

Parallel and Distributed Machine Learning Algorithms for Scalable Big Data Analytics

Henri Bal [a,1,2], Arindam Pal [b,1,2,*]

[a] Vrije Universiteit, Amsterdam, The Netherlands
[b] TCS Research and Innovation, Kolkata, India

ARTICLE INFO

Article history:
Available online 10 July 2019

ABSTRACT

This editorial is for the Special Issue of the journal *Future Generation Computing Systems*, consisting of the selected papers of the *6th International Workshop on Parallel and Distributed Computing for Large Scale Machine Learning and Big Data Analytics (ParLearning 2017)*. In this editorial, we have given a high-level overview of the 4 papers contained in this special issue, along with references to some of the related works.

Introduction

This special issue of the journal *Future Generation Computing Systems* contains four extended papers, that were originally presented at the *6th International Workshop on Parallel and Distributed Computing for Large Scale Machine Learning and Big Data Analytics (ParLearning 2017)*. The ParLearning workshops bring together the High Performance Computing (HPC) and the Artificial Intelligence (AI) and Machine Learning (ML) communities. The focus is on scaling up machine learning, data mining and reasoning algorithms from AI and ML for massive datasets, which is a major technical challenge for Big Data. In these communities, several programming environments and frameworks have been built for dealing with multithreading (e.g., OpenMP), many-cores (e.g., OpenCL, CUDA, OpenACC), Big Data (e.g., Spark, GraphLab, Hadoop), and Deep Learning (TensorFlow, PyTorch, Caffe2, and many others). ParLearning papers typically describe methods to scale up AI algorithms, using one of these frameworks, and run them on clusters, many-cores, or specialized accelerators.

ParLearning2017 was organized on May 29, 2017 at Orlando, Florida, USA, in conjunction with the 31st IEEE International Parallel and Distributed Processing Symposium (IPDPS 2017). The workshop had approximately 30 participants. We invited four of the papers presented at ParLearning 2017 for submission in expanded form to this special issue of Future Generation Computer Systems (FGCS). All papers were thoroughly reviewed, revised and improved in two rounds by 3 reviewers, after which they were accepted.

* Corresponding author.
E-mail address: arindamp@gmail.com (A. Pal).

[1] Technical Program Co-Chairs of ParLearning 2017.

[2] Guest Editors of the Special issue of Future Generation Computer Systems.

Summary of accepted papers and related works

The paper "*clMF: A Fine-grained and Portable Alternating Least Squares Algorithm for Parallel Matrix Factorization*" [1] proposes a portable Alternating Least Squares (ALS) solver for multi-cores and many-cores. The new solver optimizes thread usage and memory access and also performs architecture-specific optimizations. The new solver runs on different platforms, using OpenCL. Experiments show substantial improvements over existing solvers. A preliminary version of the paper [2] was published in ParLearning 2017. A related paper [3] studies various facets of large-scale social recommender systems, summarizing the challenges and interesting problems and discussing some of the solutions. The authors discuss the most popular techniques used in recommender systems, namely content-based filtering and collaborative filtering. They focus on large-scale recommender systems that take advantage of the characteristics of the underlying social network, variety and volatility of social bonds. They show how to tackle the problems of size and speed of change of social graphs, test the scalability of traditional recommender systems and present solutions that can take recommender systems to the next level.

The paper "*Scaling Deep Learning Workloads: NVIDIA DGX-1/Pascal and Intel Knights Landing*" [4] provides a performance and power analysis of important Deep Learning workloads on two major parallel architectures: NVIDIA DGX-1 (eight Pascal P100 GPUs interconnected with NVLink) and Intel Knights Landing (KNL) CPUs interconnected with Intel Omni-Path or Cray Aries. The evaluation consists of a cross section of convolutional neural net workloads: CifarNet, AlexNet, GoogleNet, and ResNet50 topologies using the Cifar10 and ImageNet datasets. The workloads are vendor-optimized for each architecture. Analysis indicates that although GPUs provide the highest overall

https://doi.org/10.1016/j.future.2019.07.009
0167-739X/

performance, the gap can be close for some convolutional networks; and the KNL can be competitive in performance/watt. A preliminary version of the paper [5] was published in ParLearning 2017. A related paper [6] does multi-level spatial and temporal tiling for efficient HPC stencil computation on many-core processors with large shared caches. Such platforms include DDR, the high-bandwidth RAM which is configurable either as separately addressable memory or as a large shared cache for the DDR. Examples of platforms with this feature include those containing products in the Intel® Xeon Phi™ x200 processor family (code-named Knights Landing), which use Multi-Channel DRAM (MCDRAM) technology to provide the higher bandwidth memory resources. This paper explores the application of temporal wave-front tiling to alleviate it, simultaneously leveraging both the large cache's bandwidth and the DDR capacity. Two example applications are used to illustrate the optimizations: a single-grid isotropic approximation to the wave equation and a staggered-grid formulation for earthquake simulation.

The paper "*Tera-scale Coordinate Descent on GPUs*" [7] studies the scalability of a stochastic coordinate descent algorithm. It describes a novel asynchronous implementation for GPUs that achieves much higher training speeds than the current state-of-the-art. In addition, it introduces a distributed learning system using the CoCoA framework to train larger problems that will not fit on a single node. Experiments on 16 GPUs are presented to demonstrate the scalability. A preliminary version of the paper [8] was published in ParLearning 2017. A related paper [9] proposes a method that uses high-speed rail user's information (collected by base stations) and base station information provided by the telecom operators of China to locate a high-speed train off-line. They detect an abnormal radio remote unit by using established speed models based on the gradient descent algorithm.

The paper "*Parallelizing and Optimizing Neural Encoder–Decoder Models without Padding on Multi-core Architecture*" [10] studies the scaling of Recurrent Neural Networks (RNN) based models to achieve better accuracy in Machine Translation (MT) tasks. Most implementations of Neural Machine Translation (NMT) models employ a padding strategy when processing a mini-batch to make all sentences in a mini-batch have the same length. This enables an efficient utilization of caches and GPU/SIMD parallelism but leads to a waste of computation time. They implement and parallelize batch learning for a Sequence-to-Sequence (Seq2Seq) model, which is the most basic model of NMT, without using a padding strategy. More specifically, their approach forms vectors which represent the input words as well as the neural network's states at different time steps into matrices when it processes one sentence. As a result, the approach makes a better use of cache and optimizes the process that adjusts weights and biases during the back-propagation phase. Experimental evaluation shows that their implementation achieves better scalability on multi-core CPUs. They also discuss how their approach can be used in other implementations of RNN-based models. A preliminary version of the paper [11] was published in ParLearning 2017. A related paper [12] uses stacked auto-encoder neural networks for feature extraction. The authors model high-level abstractions and reduce data dimensions by using multiple processing layers. They combine the concept of dynamic data-driven systems and stacked auto-encoders to obtain dynamic data relationships between prediction results and actual data in a dynamic environment.

Stepping into the future: Parlearning 2019

The 8th International Workshop on Parallel and Distributed Computing for Large-Scale Machine Learning and Big Data Analytics (ParLearning2019) will be held on August 5, 2019 at Anchorage, Alaska, USA, in conjunction with the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2019). The General Chairs are Arindam Pal (TCS Research and Innovation, Kolkata, India) and Henri Bal (Vrije Universiteit, Amsterdam, Netherlands). The Program Chairs are Azalia Mirhoseini (Google AI, Mountain View, CA, USA) and Thomas Parnell (IBM Research, Zurich, Switzerland). The Publicity Chair is Anand Panangadan (California State University, Fullerton, CA, USA). The Steering Committee Chairs are Sutanay Choudhury (Pacific Northwest National Laboratory, Richland, WA, USA) and Yinglong Xia (Huawei Research America, Santa Clara, CA, USA). The Keynote Speaker is Professor V.S. Subrahmanian (Dartmouth College, Hanover, NH, USA). We are looking forward to organize another successful edition of ParLearning.

Acknowledgments

We would like to thank the general chair of ParLearning 2017, Anand Panangadan (California State University, Fullerton, USA), and all the reviewers of this special issue.

References

[1] J. Chen, J. Fang, W. Liu, T. Tang, X. Chen, C. Yang, Efficient and portable ALS matrix factorization for recommender systems, Future Gener. Comput. Syst. 96 (2019) 25–38.

[2] J. Chen, J. Fang, W. Liu, T. Tang, X. Chen, C. Yang, Efficient and portable ALS matrix factorization for recommender systems, in: 6th International Workshop on Parallel and Distributed Computing for Large Scale Machine Learning and Big Data Analytics (ParLearning 2017), Orlando, Florida, USA, May 29, 2017, 2017, pp. 409–418.

[3] M. Eirinaki, J. Gao, I. Varlamis, K. Tserpes, Recommender systems for large-scale social networks: A review of challenges and solutions, Future Gener. Comput. Syst. 78 (2018) 413–418.

[4] N.A. Gawande, J.B. Landwehr, J.A. Daily, N.R. Tallent, A. Vishnu, D.J. Kerbyson, Scaling deep learning workloads: NVIDIA DGX-1/pascal and intel knights landing, Future Gener. Comput. Syst. 96 (2019) 47–57.

[5] N.A. Gawande, J.B. Landwehr, J.A. Daily, N.R. Tallent, A. Vishnu, D.J. Kerbyson, Scaling deep learning workloads: NVIDIA DGX-1/pascal and intel knights landing, in: 6th International Workshop on Parallel and Distributed Computing for Large Scale Machine Learning and Big Data Analytics (ParLearning 2017), Orlando, Florida, USA, May 29, 2017, 2017, pp. 399–408.

[6] C. Yount, A. Duran, J. Tobin, Multi-level spatial and temporal tiling for efficient HPC stencil computation on many-core processors with large shared caches, Future Gener. Comput. Syst. 92 (2019) 903–919.

[7] T.P. Parnell, C. Dünner, K. Atasu, M. Sifalakis, H. Pozidis, Large-scale stochastic learning using GPUs, Future Gener. Comput. Syst. 96 (2019) 6–24.

[8] T.P. Parnell, C. Dünner, K. Atasu, M. Sifalakis, H. Pozidis, Large-scale stochastic learning using GPUs, in: 6th International Workshop on Parallel and Distributed Computing for Large Scale Machine Learning and Big Data Analytics (ParLearning 2017), Orlando, Florida, USA, May 29, 2017, 2017, pp. 419–428.

[9] L. Ma, J. Wu, C. Li, Localization of a high-speed train using a speed model based on the gradient descent algorithm, Future Gener. Comput. Syst. 85 (2018) 201–209.

[10] Y. Qiao, K. Hashimoto, A. Eriguchi, H. Wang, D. Wang, Y. Tsuruoka, K. Taura, CaChe friendly parallelization of neural encoder-decoder models without padding on multi-core architecture, Future Gener. Comput. Syst. 96 (2019) 39–46.

[11] Y. Qiao, K. Hashimoto, A. Eriguchi, H. Wang, D. Wang, Y. Tsuruoka, K. Taura, CaChe friendly parallelization of neural encoder-decoder models without padding on multi-core architecture, in: 6th International Workshop on Parallel and Distributed Computing for Large Scale Machine Learning and Big Data Analytics (ParLearning 2017), Orlando, Florida, USA, May 29, 2017, 2017, pp. 437–440.

[12] S.-Y. Lin, C.-C. Chiang, J.-B. Li, Z.-S. Hung, K.-M. Chao, Dynamic fine-tuning stacked auto-encoder neural network for weather forecast, Future Gener. Comput. Syst. 89 (2018) 446–454.

Henri Bal heads a research group on High Performance Distributed Computing at the Vrije Universiteit in Amsterdam. He studies parallel and distributed programming systems in combination with real-world applications. His group produced programming environments such as the Orca language, MagPIe, Ibis, Satin, JavaGAT, and Swan. He is the winner of the Euro-Par 2014 Achievement Award, member of the Informatics Section of the Academia Europaea, scientific director of the ASCI research school, and coordinator of the DAS infrastructure. He is past program chair of the HPDC and CCGrid conferences and author of three books, including Modern Compiler Design.

Arindam Pal is a Research Scientist in the Embedded Systems and Robotics Group at TCS Research and Innovation. Earlier, he worked in the Data and Decision Sciences Group. He earned his Ph.D. in Computer Science from the Department of Computer Science and Engineering, Indian Institute of Technology Delhi. His broad research interests are Algorithms, Data Science, Machine Learning, Network Science, and Optimization. He has served in the organizing committee and technical program committee of several international conferences and workshops. He regularly reviews many journal papers. He has written a book chapter in the Encyclopedia of Wireless Networks. He is a Senior Member of both ACM and IEEE.